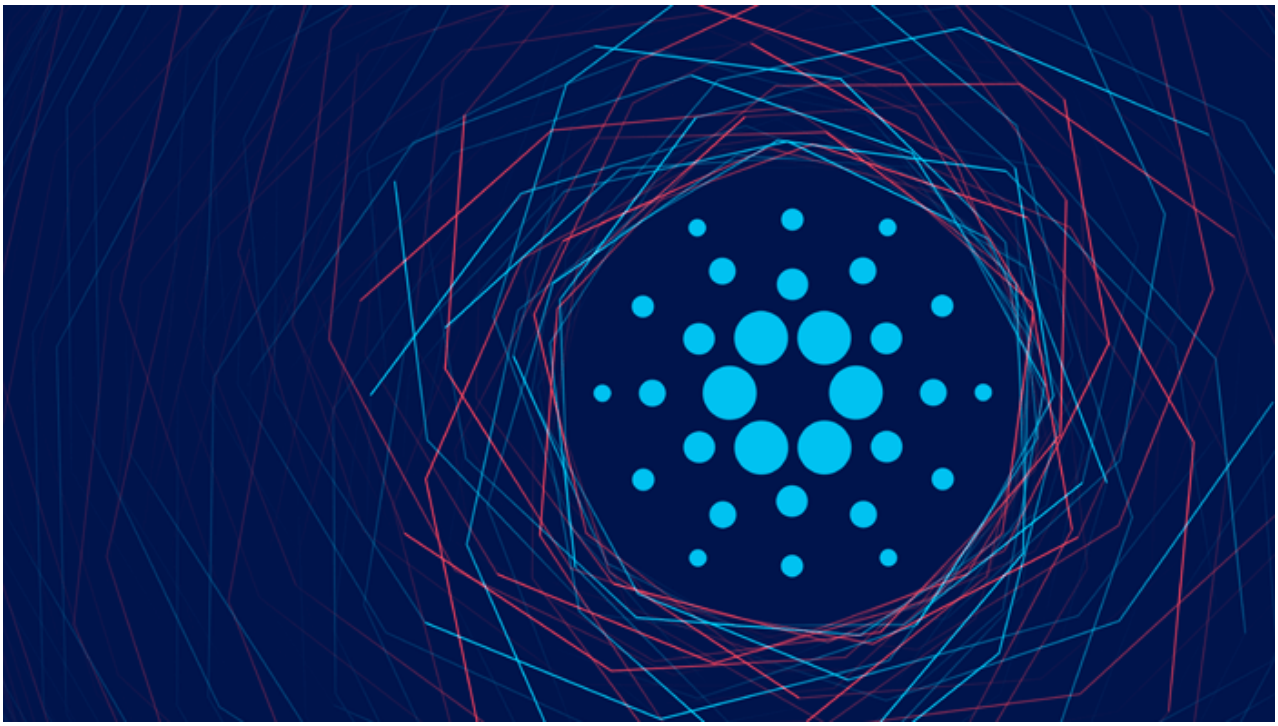
How to create Cardano Smart Contracts?

leewayhertz.com/smart-contracts-on-cardano



According to the Cardano blockchain, Cardano's new Alonzo Hard Fork gained traction as soon as it was launched on the network. The Alonzo update was released and implemented on the mainnet on the 13th of September,2021. Smart contracts can now be developed and deployed on the mainnet as a result of this development.

Alonzo provides Cardano with smart contract capabilities and enhances its functionality by incorporating the Plutus scripts written in a simple, functional language such as Solidity or Haskell and allowing users to place the scripts.

With such a beneficial update around smart contracts on Cardano, we must learn about smart contract development on the Cardano platform. To begin with, let's understand the basic definition of smart contracts.

What are Smart Contracts?

Smart contracts are pre-programmed, automatic digital agreements. They are self-executing and unchangeable. They do not entertain any activity of a middleman or the presence of any third parties.

We can divide smart contracts into two categories:

In one scenario, you want to insert a value concept from one actor (or set of players) to another actor (or a group of actors). There must be a representation of that value, as well as the rules and circumstances that govern it, as well as an event to activate it. This is

referred to as a financial contract, and it is best executed using a domain-specific language.

In the other scenario, you want to develop a program and application. This application is made up of a triangle:

- The client is the program that runs on your computer.
- The server is what operates on someone else's server (or multiple servers).
- The smart contract is the part of code that allows decentralized operations to take place.

What programming languages does Cardano use for its smart contract development?

There are three specific languages for smart contract development on Cardano, as mentioned below:

Marlowe

Marlowe is a domain-specific language (DSL) that enables users to create blockchain applications specifically targeted at financial transactions. When compared to a Turing-complete language, the Marlowe DSL provides:

- Better Security
- Assured Certainty
- Guarantee of termination
- Much better guarantee of the correctness of behavior

The following features are guaranteed by the design of Marlowe:

- Contracts have a defined duration, and there is no recursion or looping
- Contracts will come to an end, and all actions are subject to a timeout
- Contracts have a set length of time to be effective
- At the time of closing, no assets are kept
- Preserving value

Plutus

Plutus is the Cardano blockchain's smart contract platform. It enables the development of apps that communicate with the Cardano blockchain. Plutus enables all programming to be done in Haskell using a single library. It enables the development of safe apps, new asset acquiring, and construct smart contracts in the most predictable, deterministic environment possible. Additionally, developers are not required to test their work on a complete Cardano node. You can do the following with Plutus:

- Create fresh tokens in a minimal environment
- Construct smart contracts
- Support for simple multi-signature scripts

Haskell

Haskell is the fundamental language for Plutus. It is a programming language used by Cardano for its smart contract creation. Haskell also regulates Marlowe, a domain-specific language for creating Cardano's financial smart contracts. Even with not so high ranking on google, Haskell is the first choice of Cardano when it comes to implementing a programming language. Why does Cardano do so?

Let's understand the motto of Cardano hidden behind choosing Haskell in the first place. The basic explanation for this is that Haskell has the capability and power to write appropriate codes which are robust. Haskell was given its name based on the name of a famous American mathematician known as Haskell Curry. Curry had his roots in the field of functional programming languages, for instance, Miranda. His interest in functional programming languages laid the ground for defining Haskell in 1990.

Haskell is thus a functional programming language that appropriately creates high-assurance codes that need a relevant degree of verification of formal nature. As Haskell provides an elevated degree of certainty, this helps the Cardano developers ensure that the implemented code is robust and correct.

How to create Cardano smart contracts?

Cardano uses Marlowe and its six distinct steps of creating a smart contract. The eight steps are:

- Pay
- Close
- Values, Observations and Actions
- Oracles
- If
- When
- Let
- Assert

At every step in the execution process, along with going back to a new state and the continuation of a contract, there is a chance that it affects the payments and creates warnings. To explain these contracts, the Marlowe values-observations and actions are required for supplying the external information.

Pay

A payment agreement Pay a p t v cont will transfer the value v of the token t from the account a to the payee p, which will be one of the contract participants or another account in the contract. If the value of v is negative, or if there are insufficient funds to complete the payment in full, warnings will be produced (even if there are positive balances of other tokens in the account). In the latter instance, a partial payment is made (of all available funds). The continuation contract is indicated in the contract by the term cont.

Close

The term Close specifies how the contract will be canceled (or terminated). Its sole function is to reimburse owners of accounts with a positive balance. This process is repeated for each account, but all accounts are reimbursed in a single transaction. Before delving into additional types of contracts, it's necessary to define values, observations, and actions.

Values, observations and actions

The term "values" refers to some numbers that vary over time, such as "the current slot number," "the current balance of some token in an account," and any previously made decisions; these are referred to as volatile values. Additionally, values may be mixed with addition, subtraction, and negation and conditional on an observation.

Observations are Boolean values that are obtained by value comparison and maybe merged using normal Boolean operators. Additionally, one may examine if a choice has been taken or not for the Boolean values.

At each stage of execution, observations will have a value. In contrast, actions occur at certain moments throughout execution. As previously stated, acts can be:

- money depositing,
- selecting one of several possibilities, including an oracle value (see the next section), or
- indicating some external worth.

Oracles

Oracles are being created for the Cardano blockchain as a whole and will be accessible for users in Marlowe on Cardano. Oracles are modeled as decisions made by a participant with a specialized Oracle role, "Kraken."

If a contract's role is "Kraken," and that role makes an option like "dir-adausd," the Playground simulation will pre-fill this choice with the current value of the direct ADA/USD conversion rate based on data from Cryptowat.ch. It is also possible to acquire the inverse rates of the currency pairings mentioned by replacing the prefix inv- with the prefix inv-.

If

If the conditional is true, If obs cont1 cont2 is performed, it will continue as cont1 or cont2, based on the Boolean value of the observation obs.

When

With the form When cases timeout cont., this is the most complicated contract function Object() { [native code] }. It is a contract triggered by activities that may or may not occur at any given time: the cases in the contract explain what occurs when certain actions occur.

According to the contract, When cases timeout cont, a collection of cases is added to the list cases. Each case takes the form Case ac co, where ac denotes action and co denotes a continuation (another contract). When a certain action, such as ac, occurs, the state changes and the contract continues as the appropriate continuation co.

To ensure that the deal finally moves forward, the case timeout cont will continue as cont when the timeout and a slot number are reached.

Let

A lease agreement The let id Val cont function allows a contract to name a value with an identifier. The expression value is evaluated and saved with the name id. The contract is then extended as cont.

This technique not only allows us to utilize abbreviations, but it also allows us to capture and preserve volatile data that may change over time, such as the current price of oil or the current slot number, at a specific moment in contract execution, to be used later in contract execution.

Assert

A contract that asserts Assert obs cont does not affect the contract's state; it immediately continues as cont, but it provides a warning if the observation obs is untrue. It may be used to guarantee that a property holds at any point in the contract because the static analysis will fail if any execution results are in a false assert.

What makes the Cardano blockchain stand out from other blockchains?

Cardano is designed to be scalable, sustainable, and interoperable with other blockchains and system architectures.

In comparison to other blockchain protocol initiatives, Cardano is distinct in several ways. For example, Cardano protocol development is based on peer-reviewed research, high-assurance code is used at the highest levels of engineering, and the protocol is developed using Haskell as a functional language.

Cardano smart contracts must be written in Plutus or IELE, designed to provide a better level of certainty. Plutus is a smart contract language written in Haskell. Haskell is well-known among academics and developers for its combination of academic and industry-grade expertise with fundamental computer science qualities and codes. Thus, creating smart contracts on the Cardano platform will be more secure and trustworthy than writing smart contracts in any other smart contract language. Plutus Platform is based on the Haskell framework that will serve as an accessible toolkit for developers to construct smart contracts. It will also enable both on-chain and off-chain code. Cardano's smart contract code is safe, tested, and documented due to peer review and high assurance.

Finally, this research-first strategy taken by a properly credentialed team of academics and cryptography specialists distinguishes Cardano and Cardano smart contracts from its competitors.

Cardano's future strength lies in its capacity to operate as a binding and trustworthy entity to transfer shareholder assets. These stakeholder assets are necessary for the contracting parties to participate. The contract's assets will be transferred following a set of rules agreed upon by the parties and programmed into the contract. However, monies pledged to a smart contract will never be "frozen" indefinitely. The writers can implement a timeout to ensure that money is repaid after a certain span of time.

A smart contract created and programmed in Plutus on the Cardano blockchain provides complete visibility to all parties engaged in the contract. When constructed correctly, a single hostile actor can't engage.

Use Cases of Cardano blockchain

Crowdfunding

Crowdfunding has evolved as an egalitarian, dispersed method of obtaining startup cash. This seed cash, which is typically donated in modest sums by many individuals, enables a project to be built to completion. Once the project has received sufficient funding, the final product is developed and given to the financial contributors. If the product does not receive full funding, the funds generated are returned to the financial supporters. This is a popular method of financing that circumvents typical venture capital or startup loans.

A smart contract built on the Cardano blockchain may be used to represent crowdfunding campaigns with transparency and the assurance that funds would be refunded to donors if certain conditions are not satisfied. In each of these instances, the contract is influenced by time (limited-time fundraising drive) and the acts of others (sending value). If the project's goal is not fulfilled after a specified period of time, the money is refunded to the backers. If the fundraising target is attained, the funds will be sent to the project's authors.

NFT Marketplace

With various updates on the Cardano platform, you can also develop your NFT Marketplace on the Cardano blockchain platform. NFT Marketplace built on Cardano is realistic with reliable security, efficiency and high throughput. You can get your custom NFT Marketplace on Cardano. Using Cardano, you can develop NFT Marketplace with advanced technologies and integrations of third parties matching the market trends. Cardano has a customer-centric approach that helps create a user-friendly NFT Marketplace regulating seamless trade and creation of NFTs.

dApp Development

One of the real-world applications of Cardano is decentralized applications (dApps). Cardano has various categories of dApps like DeFi, voting, identity management, games, etc. Cardano dApps are powered by the Cardano smart contracts, which has robust codes written in Haskell. Cardano smart contracts help create quick, reliable and highly assured dApps with the assistance of Haskell and Plutus. These dApps have a user-interactive interface with Cardano blockchain and execute transactions without letting in any third party.

Ending Note

Contracts and value-related agreements affect our financial environment profoundly. Cardano smart contracts will provide a very effective digital platform for modeling and executing real-world contracts. When created on the Cardano blockchain, these contracts provide complete visibility to all contract parties while also being very safe and self-executing according to the contract's set requirements. Developers may utilize the Plutus Platform to develop effective methods for securely transferring value and providing services to many individuals globally.

If you are looking for smart contract developers, you are at the right place. Connect with our smart contract developers for further guidance.